Day 10: Practical Social Media Data Mining

Kenneth Benoit and Slava Mikhaylov

Introduction to Data Science and Big Data Analytics

26 August 2016

Day 10 Outline

Practical Tools for Data Science

Social Media Data

Practical Tools for Data Science

Tools for Data Science: Categories of tools

- Programming languages
- Specialised libraries and packages within programming languages
- Graphical interfaces and visualization
- Online communities and code hosting, source control
- Big data, databases and parallelization
- Tools specifically for text analysis

Programming languages

- Statically typed:
  - C, C++, Java, Fortran, c
  - Slower to learn, slower to write in, faster to run
- Dynamically typed ('scripting')
  - : perl, python, ruby, javascript, php
- Data analysis
  - : R, matlab, octave

Data analysis software

- R and python currently most popular for data science
- Python: Pandas, NLTK, Scikit-learn
- R: CRAN and many small, custom packages
- Java: WEKA
- Graphical interfaces: Stata, SAS, SPSS

Systems, Text editors and IDEs

- Windows, Mac, or Linux
- vim and emacs
- notepad++, textmate, gedit, sublimetext, atom
- R: RStudio, RCommander, Revolution R
- Java: Eclipse, Netbeans
- Python: Spyder, PyCharm

Visualization

- R: ggplot2
- Python matplotlib
- Gephi: network visualization `http://gephi.github.io/`

Code hosting and source code management

- Version control/Source code management: git, mercurial, svn
- Hosting: github, bitbucket, gitlab
- github also serves as a community for discussion and collaboration

Resources for learning to code

- http://tryr.codeschool.com/
- https://www.coursera.org/course/programming1
- https://developers.google.com/edu/
- http://stackoverflow.com/
- http://stats.stackexchange.com/
- http://www.google.com/advanced_search

Big data and parallel and distributed processing

- ‘Big Data’ ill-defined, but should refer to data that can’t be processed in-memory on local machine
- This implies a changing definition
- Parallel processing: R package ‘parallel’
- Distributed processing: MapReduce with Hadoop, Apache spark
- Cloud computing
- Databases: Relational (e.g. SQL,) non-relational ’NoSQL’ (e.g. redis, cassandra)

Packages for quantitave text analysis

- Stanford CoreNLP, Mallet (Java)
- NLTK, gensim, TextBlob (Python)
- tm, quanteda (R)
- Alceste, WordStat (QDA Miner)
- Nvivo, atlas.ti

Books for text analysis

- Natural Language Processing with Python - NLTK
- Foundations of Statistical Natural Language Processing (Manning and Schutze)
- Introduction to Information Retrieval

Handling text data

- Usually less data-intensive than image, audio, or video
- ASCII, UTF-8: 1 byte per character ($2^7 = 128$ chars)
- E.g., entire proceedings of European Parliament, 1996-2005, in 21 languages [1]: 5.4GB
- Often the difficulty is $p >> n$, rather than 'big data'.

Scraping text from the web

- web crawlers/spider download sites by traversing links
- Python - scraPy, Beautiful Soup
- R - Rvest
- Chrome web plugins, import.io
- cUrl, wget, or other tools available ('httrack')
- Problems: rate limiting, ethical issues

Make scraping unnecessary!

- Organizations and governments should be aware of need for open, machine-readable data
- data.gov.uk, data.gov
- Data should be available in human and machine format!
- Make the raw data available in as many formats as possible.
- Consider machine readability at time of data collection
- Provide an Application Programming Interface (API)

Social Media Data

Why social media data?

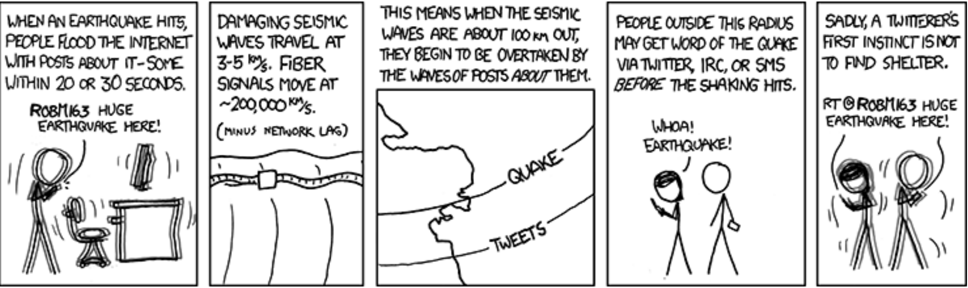
- Volume and coverage
- Twitter: 316 million monthly active users, 500m tweets per day [2]
- Facebook: 968 million daily active users on average for June 2015, 1.49 billion monthly active users as of June 30, 2015 [3]
- Real time — new data is available (somewhat) publicly immediately on current events
- Metadata — geographic location, user device, profile, timestamp and other metadata is accessible.

Why social media data?

- Good case for machine learning and data mining — lots of data, lots of metadata
- Many-to-many *broadcast* text corpus
- Social network analysis: a graph of social connections
- Passively sampling has some advantages over other survey methods

- Broadcast
  - simplex (e.g. radio, semaphore, smoke signal)
  - duplex (e.g. round-table meeting)
  - publish-subscribe (e.g. twitter, mailing list)
- Point-to-point: sender specifies receivers
- Social media allow many of these different forms of communication
- Twitter is broadcast publish-subscribe
- Every user is a sensor: receiver and broadcaster — a distributed sensor network [4]
- `https://www.youtube.com/watch?v=XJ1EQbmJ_LQ`

Seismic Waves



Why not?

- Legal, ethical and privacy concerns:
- twitter is relatively public, facebook relatively private
- legal issues need to catch up with the technology
- Are EULAs (End-User License Agreement) too complex to allow ‘informed consent’?
- A/B testing doesn’t need consent, should social experiments? ($N = 689{,}003$)[5]

Why not?

- Unconventional language use — slang, txtspk, emoticons :-(
- Sampling issues and many new methodological headaches: homographs
- Very difficult to interpret tweet frequencies: What causes someone to tweet?
- Biased sample (Barbera and Rivero 2013)
- commercial interfaces are brittle, inconsistent, opaque and present at challenge for replication

Privacy: example

- Example: FOI request for NYC taxi fare logs:
- Medallian numbers and cab numbers were mapped to unique ids with MD5 hash
- ids follow certain pattern, only 22M possible
- Can compute all hashes in 2 minutes [6]

Example applications

- Tracking disease (ILI) through search terms and social media (Lampos et al 2015)
  - Geo-located tweets for 154 weeks
  - Manual list (dictionary) of 36 ngrams (up to 4grams) that were associated with illness, expanded to 205 with co-occurrence matching
  - To measure level of illness and association with vaccination program
  - Regularized linear regression, X is ngrams frequencies, Y is health statistics

Example applications

- Predicting election outcomes or polls
- Sentiment: particularly for financial or corporate interests
- (Vasileios Lampos: www.lampos.net)
- Government security/intelligence
- Social network analysis: a graph of social connections

How can we access this data?

- API: Application Programming Interface — a way for two pieces of software to talk to each other
- Twitter, facebook, google — all expose public web services
- Your software can receive (and also send) data automatically through these services
- Data is sent by `http` — the same way your browser does it
- Most services have helping code (known as a wrapper) to construct http requests
- both the wrapper and the service itself are called APIs
- http service also sometimes known as REST (REpresentational State Transfer)

HyperText Transfer Protocol

Anatomy of a http request

`https://api.twitter.com/1.1/search/tweets.json?`
`q=Nick+Clegg%21&since_id=24012619984051000&max_id=25012619`

`Nick Clegg!` becomes `Nick+Clegg%21`

- Parameters to the API are encoded in the URL
- you must encode requests — spaces and non ASCII characters are replaced

Available social media APIs

- Wikipedia: mediawiki
- Google
- google plus
- blogger
- reddit
- foursquare
- facebook
- twitter: 'Gardenhose' (REST, Streaming), firehose, commercial

The twitter APIs: REST

- This is the most comprehensive API
- Returns a sample of historical data from the last 8–10 days.
- Stateless: you send a command and receive a result.
- http GET requests return information
- http POST requests upload or alter information (e.g. twitterbots)
- The manual: `https://dev.twitter.com/rest/public`
- R package : twitteR

The twitter APIs: Streaming

- Connect to the twitter server and collect tweets as they fly by.
- The manual:
  `https://dev.twitter.com/streaming/public`
- R package: streamR

Authentication

- Username and Password
- Oauth (ROauth): share a key without sharing a username and password
- IP address limitations
- Rate limitations
- Per-user and per-application

Other options

- The firehose: work with twitter
- Commercial options: GNIP (now bought by twitter) and Datasift

The Output: JSON and XML

- XML: eXtensible Markup Language: encodes documents in a form that is both human-readable and machine readable
- JSON : JavaScript Object Notation
- If you have a choice, you probably want JSON
- JSON uses key:value pairs, XML uses trees
- JSON is easily read into a programming language
- Sometimes known as serialization formats

And finally... the data.

- Full of spam, bots, unicode, and gibberish
- Homographs and ambiguities are a problem, e.g. Clegg, Cameron, Miliband
- Lots of retweets (approximately one-third retweets, replies, tweets)
- Only 1% show location — some methods exist to infer location
- All aspects of metadata and reply/retweet structure are available
- All aspects of network structure: followers and ‘friends’, profile information

Twitterbots

- API also allows actions such as posting tweets (POST)
- Examples:
- @netflix_bot posts new content using netflix api
- @eqbot posts earthquake warnings
- @pentametron posts pairs of tweets in rhyming couplets [7]

Twitterbots


Big Ben
@big_ben_clock

BONG BONG BONG BONG BONG BONG BONG BONG BONG BONG

10:00 AM - 10 Oct 2014

73 63